

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-079290

(43)Date of publication of application : 22.03.1996

(51)Int.Cl.

H04L 12/437

(21)Application number : 06-208306

(71)Applicant : FUJITSU LTD

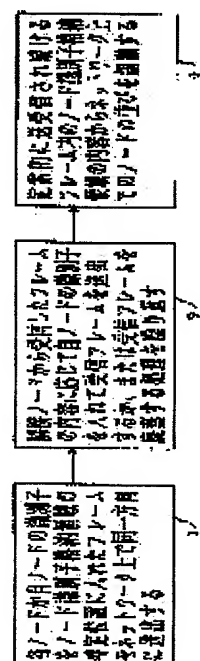
(22)Date of filing : 01.09.1994

(72)Inventor : NAKAZUMI MASASHI

(54) METHOD FOR DETECTING COMMUNICATION NETWORK CONSTRUCTION AND GENERATING LINE INFORMATION TABLE**(57)Abstract:**

PURPOSE: To provide a network constitution detecting method for automatically detecting the line of nodes by each node composing a communication network and an automatic generating method of a line information table in which the transmission origin node and the reception destination node of each channel are set.

CONSTITUTION: The frame in which each node connected in a ring shape puts its own node identifier in the specified location of a node identifier storage area is transmitted in the same direction 1. A processing in which its own node identifier is stored in the frame received from an adjacent node is stored and the received frame is retransmitted or a received frame is aborted is repeated 2. The row of nodes on a network is recognized by each node according to the contents of the node identifier storage area within the frame transmitted and received normally on the network 3.

**LEGAL STATUS**

[Date of request for examination] 30.07.1999

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3307508

[Date of registration] 17.05.2002

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-79290

(43) 公開日 平成8年(1996)3月22日

(51) Int.Cl.⁶

識別記号

庁内整理番号

F I

技術表示箇所

H 0 4 L 12/437

H 0 4 L 11/ 00

3 3 1

審査請求 未請求 請求項の数13 O L (全 16 頁)

(21) 出願番号 特願平6-208306

(22) 出願日 平成6年(1994)9月1日

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72) 発明者 中住 誠志

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(74) 代理人 弁理士 大菅 義之 (外1名)

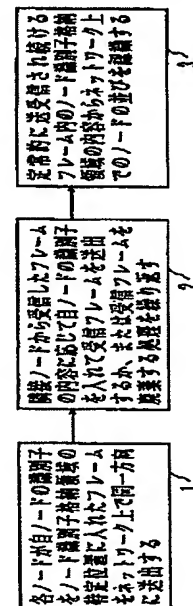
(54) 【発明の名称】 通信ネットワーク構成検出および回線情報テーブル生成方法

(57) 【要約】

【目的】 通信ネットワークを構成する各ノードがノードの並びを自動的に検出するためのネットワーク構成検出方法と、各チャネルの送信元ノードと受信先ノードとが設定される回線情報テーブルの自動生成方法を提供することを目的とする。

【構成】 リング状に接続された各ノードが自ノード識別子をノード識別子格納領域の特定位置に入れたフレームを同一方向に送出し、隣接ノードから受信したフレームに自ノード識別子を格納して再送出するか、または受信フレームを廃棄する処理を繰り返し、定常的にネットワーク上で送受信されるフレーム内のノード識別子格納領域の内容によってネットワーク上でのノードの並びを各ノードが認識する。

本発明の第1の実施例に対する
機能ブロック図



【特許請求の範囲】

【請求項 1】 複数の通信ノードがリング状に接続された通信ネットワークにおいて、

該複数の各ノードが、自ノードの識別子をノード識別子格納領域の特定位置に入れたフレームを前記リング状の通信ネットワーク上で同一の方向に送出し（１）、
該複数の各ノードが、該通信ネットワーク上で隣接ノードから受信したフレームの内容に応じて該受信フレームのノード識別子格納領域に自ノードの識別子を格納して、前記同一の方向に再送出するか、または受信フレームを廃棄する処理を繰り返す（２）、
該通信ネットワーク上で定期的に送受信され続けるフレーム内のノード識別子格納領域の内容によって、該通信ネットワーク上でのノードの並びを該複数の各ノードが認識する（３）ことを特徴とする通信ネットワークにおけるネットワーク構成検出方法。

【請求項 2】 前記ノード識別子格納領域の特定位置が該領域の先頭であることと、
前記受信フレームの再送信または廃棄の処理に先立って、受信フレーム内に自ノードの識別子がすでに格納済みであるか否かを判定し、すでに格納済みである時には該受信フレームへの自ノードの識別子の格納、または該フレームの廃棄を行うことなく該フレームを同一方向に再送出することを特徴とする請求項 1 記載の通信ネットワークにおけるネットワーク構成検出方法。

【請求項 3】 前記受信フレーム内に自ノードの識別子が格納済みでない時、該受信フレーム内のノード識別子格納領域の先頭に格納されているノード識別子が自ノードの識別子より大きいかなんかを判定し、大きい時には該受信フレームを廃棄することを特徴とする請求項 2 記載の通信ネットワークにおけるネットワーク構成検出方法。

【請求項 4】 前記先頭の識別子が自ノードの識別子より大きいかなんかの判定において自ノードの識別子の方が大きい時、該受信フレーム内のノード識別子格納領域の格納済識別子の最後に自ノードの識別子を格納して、前記同一方向に該受信フレームを再送出することを特徴とする請求項 3 記載の通信ネットワークにおけるネットワーク構成検出方法。

【請求項 5】 前記受信フレーム内に自ノードの識別子が格納済みでない時、該受信フレーム内のノード識別子格納領域の先頭に格納されているノード識別子が自ノードの識別子より小さいかなんかを判定し、小さい時には該受信フレームを廃棄することを特徴とする請求項 2 記載の通信ネットワークにおけるネットワーク構成検出方法。

【請求項 6】 前記先頭の識別子が自ノードの識別子より小さいかなんかの判定において自ノードの識別子の方が小さい時、該受信フレーム内のノード識別子格納領域の格納済識別子の最後に自ノードの識別子を格納して、前

記同一方向に該受信フレームを再送出することを特徴とする請求項 5 記載の通信ネットワークにおけるネットワーク構成検出方法。

【請求項 7】 前記同一の方向として右回りを採用した場合と、左回りを採用した場合との前記ノードの並びの認識結果の比較によって、該認識結果の正当性をチェックすることを特徴とする請求項 1 記載の通信ネットワークにおけるネットワーク構成検出方法。

【請求項 8】 前記定期的に送受信され続けるフレームの内容を前記複数の各ノードがモニタし、該定期的に送受信され続けるフレームが常に同一でない時に該通信ネットワーク上に同一識別子を有するノードが複数存在すると判定することを特徴とする請求項 1 記載の通信ネットワークにおけるネットワーク構成検出方法。

【請求項 9】 前記同一の方向として右回りを採用した場合と左回りを採用した場合とについて、両方向の隣接ノードから受信したフレームのうち、片方の方向から受信したフレーム内の前記ノードの並びを自ノードの識別子を先頭にして並べ替え、

該並べ替えの結果と前記片方の方向以外の方向から受信したフレーム内のノード識別子の並びが異なる時に、該通信ネットワーク上に同一識別子を有するノードが複数存在すると判定することを特徴とする請求項 1 記載の通信ネットワークにおけるネットワーク構成検出方法。

【請求項 10】 複数のノードによって構成される通信ネットワークにおいて、

該通信ネットワーク内で通信が行われている回線の送信元ノードが該送信元ノードを示すソースアドレスを格納したソースフレームをネットワーク内で該通信の受信先ノードの方向に送出し（６）、
該受信先ノードが該受信先ノードを示すディスティネーションアドレスを格納したディスティネーションフレームを該送信元ノードの方向に送出し（７）、
該送信元ノードと受信先ノード、および該 2 つのノード間の中継ノードが、該ソースフレームとディスティネーションフレームに格納されたソースアドレスとディスティネーションアドレスとに基づいて、該回線に対応して送信元ノードと受信先ノードとを示す情報を回線情報として格納する回線情報テーブルを生成する（８）ことを特徴とする回線情報テーブル生成方法。

【請求項 11】 前記通信が行われている回線の送信元ノード、または受信先ノードにおける前記回線情報テーブル生成処理において、

自局のみで判明する回線情報を設定し、
ネットワークを介してフレームを受信し、該受信フレームのうちで自ノードが送信側として送るべきフレーム内にソースアドレスとして、また自ノードが受信側として折り返すべきフレーム内にディスティネーションアドレスとして、共に自ノードのアドレスを格納した送信フレームを作成し、

該送信フレームと前記受信フレーム内の対応個所のデータを比較して不一致がある時、該送信フレームを送信して、前記ネットワークを介したフレーム受信以降の処理を繰り返す、

該対応個所のデータを比較して一致した時、該当回線に対する回線情報テーブルの作成処理のためのフレーム送受信を終了することを特徴とする請求項 10 記載の回線情報テーブル生成方法。

【請求項 12】 前記通信が行われている回線の送信元ノードと受信先ノードとの間の中継ノードにおける前記回線情報テーブル生成処理において、自ノードが該回線に対して中継ノードであることを示すスルーに設定されていることを確認し、回線上のフレームをモニタして各回線のソースアドレスとディスティネーションアドレスとを抽出し、前記回線情報テーブルを生成することを特徴とする請求項 10 記載の回線情報テーブル生成方法。

【請求項 13】 前記ソースフレームまたはディスティネーションフレームとして、前記通信ネットワーク内で通信に用いられているフレームの余剰ビットに前記ソースアドレスまたはディスティネーションアドレスを格納したフレームを用いることを特徴とする請求項 10 記載の回線情報テーブル生成方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は通信ネットワークにおける通信方式に係り、さらに詳しくは通信ネットワーク上でのノードの並びを検出するネットワーク構成検出方法と、通信ネットワークにおける各チャネルの送信元ノードと受信先ノードとが設定される回線情報テーブルの自動生成方法に関する。

【0002】

【従来の技術及び発明が解決しようとする課題】 通信ネットワークにおけるネットワークのトポロジや、各通信回線、すなわちチャネルがどのノードからどのノードに向かって設定されているかなどのデータは一般にそれぞれテーブルとして各ノードに備えられている。通信ネットワークにおけるノードの並び、すなわちトポロジが設定されるトポロジテーブル、および回線（チャネル）の送信元ノードと受信先ノードとが設定される通信回線情報テーブルの内容は従来人手によって設定され、管理されていた。そのため、例えば回線情報テーブルの内容が実際のネットワークにおける通信状態と一致していなくても、その不一致を各ノードが容易に検出できないと言う問題点があった。

【0003】 本発明は、例えばリング状の通信ネットワークにおいて、通信ネットワークを構成する各ノードがネットワークの構成、すなわち各ノードの並びを自動的に検出する方法と、通信ネットワーク内での各チャネル

の送信元のノードと受信先のノードとが設定される回線情報テーブルを自動的に生成する方法を提供することを目的とする。

【0004】

【課題を解決するための手段及び作用】 本発明においては、第 1 の実施例として複数の通信ノードがリング状に接続された通信ネットワークにおけるノードの並びを各ノードが認識するための通信ネットワーク構成検出が行われる。

【0005】 図 1 はこの第 1 の実施例、すなわち通信ネットワーク構成検出方法の機能ブロック図である。同図において、1 で通信ネットワークを構成する複数の各ノードが、自ノードの識別子をノード識別子格納領域の特定位置、例えば先頭に入れたフレームをリング状の通信ネットワーク上で同一の方向、例えば右回り方向に送出する。

【0006】 続いて、2 で複数の各ノードは、通信ネットワーク上で隣接ノードから受信したフレームの内容に応じて、その受信フレーム内のノード識別子格納領域に自ノードの識別子を格納して前述の同一方向に再送出するか、または受信フレームを廃棄する処理を繰り返し、3 で通信ネットワーク上で定期的に送受信され続けるフレーム内のノード識別子格納領域の内容によって通信ネットワーク上でのノードの並びを認識する。

【0007】 本発明の第 1 の実施例においては、通信ネットワークを構成する各ノードはノード識別子格納領域の特定位置、例えば先頭に自ノードの識別子を入れたフレームを例えば右回り方向に送出する。そして、左方向の隣接ノードから受信したフレーム内に自ノードの識別子がまだ格納されていない場合には、そのフレームの先頭にすでに格納されているノード識別子が自ノードの識別子より大きいとか否かを判定し、大きい時にはその受信フレームを廃棄する。

【0008】 これに対して、先頭に格納されている識別子の方が小さい時には、自ノードの識別子を受信フレームのノード識別子格納領域の格納済識別子の最後に格納し、そのフレームを右回り方向に再び送出する。

【0009】 このような処理が各ノードで繰り返されることによって、ノード格納領域の先頭に最小の識別子が格納されたフレームのみが定期的にネットワーク内で送受信され続けることになり、このフレームの内容から各ノードはノードの並び、すなわちリングトポロジテーブルの内容を設定することが可能となる。

【0010】 本発明の第 2 の実施例においては、複数の通信ノードから構成される通信ネットワークにおいて、通信が行われている回線に対応して、その回線の送信元ノードと受信先ノードとを回線情報として格納する回線情報テーブルの生成が行われる。

【0011】 図 2 はこの第 2 の実施例、すなわち回線情報テーブル生成方法の機能ブロック図である。同図にお

いて、6で通信が行われている回線の送信元ノードが、例えば通信に用いられているフレームの余剰ビットにその送信元ノードを示すソースアドレスを格納したソースフレームをネットワーク内で通信の受信先ノードの方向に送出する。

【0012】続いて7で通信の受信先ノードが、例えば通信に用いられているフレームの余剰ビットに受信先ノードを示すディスティネーションアドレスを格納したディスティネーションフレームを送信元ノードの方向に送出する。

【0013】回線の送信元ノードと受信先ノード、およびこれら2つのノードの間の中継ノードは、ソースフレームとディスティネーションフレームに格納されたソースアドレスとディスティネーションアドレスとに基づいて、回線に対応して送信元ノードと受信先ノードとを回線情報として格納する回線情報テーブルをそれぞれ生成する。

【0014】第2の実施例においては、例えばノード1からノード2に向かってE-W方向の回線、ノード2からノード1に向かってW-E方向の回線が張られている場合には、ノード1からソースアドレスが格納されたE-Wソースフレームがノード2にE-W回線を介して送られ、またノード2からノード1に対してノード2のアドレスをディスティネーションアドレスとして格納したE-Wディスティネーションフレームが送られ、それぞれ相手側のノードでこれらのアドレスが取り出されることにより、E-W方向の回線に対するソースノードとディスティネーションノードとを、ノード1とノード2の両方が認識することが可能となる。

【0015】以上のように本発明によれば、通信ネットワーク上でのノードの並びを示すネットワーク構成の検出と、通信に用いられている回線の送信元ノードと受信先ノードとを示す回線情報テーブルの生成が自動的に行われる。

【0016】

【実施例】まず、本発明の第1の実施例としてネットワーク内のノードの並びを検出するネットワーク構成検出方法について詳細に説明する。図3は第1の実施例におけるリングトポロジテーブルの構築処理フローチャートである。同図は複数の通信ノードがリング状に接続されたネットワークにおいて、ノードの並びを各ノードが認識することを可能とする通信ネットワーク構成検出方法の処理手順を示す。

【0017】図3において処理が開始されると、まずステップS1において各ノードは自ノードの識別子をノード識別子格納領域の先頭、すなわち第1バイト目に入れたフレーム（識別子格納領域の残りにはFFを格納する）を右方向に送出し、ステップS2で左方向からのフレームを受信する。そして、ステップS3で受信フレーム内に自ノードの識別子がすでに格納されているか否か

を判定する。

【0018】自ノードの識別子が格納されていない場合には、ステップS4でその受信フレームの第1バイト目に格納されているノード識別子が自ノードの識別子より大きいかなかを判定し、大きい時には受信フレームを廃棄してステップS2以降の処理を繰り返す。

【0019】これに対して、受信フレームの第1バイト目に格納されている識別子が自ノードの識別子より小さい時には、ステップS6で受信フレームのノード識別子格納領域内の格納済識別子の最後に自ノードの識別子を追加し、そのフレームを再び右方向に送出し、ステップS2以降の処理を繰り返す。

【0020】ステップS3で受信フレームの内部に自ノードの識別子がすでに格納されていると判定された場合には、以前にステップS6で自ノードの識別子を格納したフレームを送出してから、そのフレームが通信ネットワーク内を一周して自ノードに戻ってきたことを意味し、この一周したフレームはその後通信ネットワーク内で定常的に送受信され続けることになるため、リングトポロジテーブル構築処理は終了する。

【0021】図4は図3の処理フローチャートを用いたリングトポロジテーブル構築例の説明図である。図4においてリング状に接続された複数の通信ノード、ここではノード1からノード4がそれぞれノード識別子格納領域の先頭、すなわち第1バイト目に自ノードの識別子を入れたフレームをそれぞれ、例えば時計回り（右回り）に隣接ノードに向けて送出する。第2バイト目以降には、例えばFFを挿入したフレームを送出する。

【0022】このフレームを受け取った各ノードは、それぞれノード識別子格納領域の第1バイト目に格納されているノード識別子が自ノードの識別子より大きい場合には受信フレームを廃棄し、自ノードの識別子より小さい場合にはノード識別子格納領域内で格納済識別子の最後に自ノードの識別子を格納したフレームを、再び例えば時計回りに隣接ノードに向かって送出する。

【0023】例えば、ノード3では、ノード1から受け取ったフレームの第2バイト目に自ノードのノード識別子“03”を格納して、ノード2に向かって送出する。そして、このフレームがノード2、ノード4、ノード1を経由して再び自ノードに戻ってきた時点で、その受信フレームに自ノードの識別子が格納されているために、これが定常的に送受信されるフレームであると認識し、このフレームのノード識別子格納領域の内容によってリングトポロジテーブルの設定内容を認識する。

【0024】ノード2においては、隣接ノードとしてのノード3から最初に送出されたフレームに対しては、その第1バイト目のノード識別子が自ノードの識別子より大きいためにそのフレームを廃棄する。そして、ノード1からノード3を経由して送られたフレーム、すなわち第1バイト目に“01”が格納され、第2バイト目に

“03”が格納されたフレームに、自ノードの識別子“02”を追加して隣接ノード4に対して送出する。このフレームは、隣接ノード4においてノード識別子格納領域の最後にノード識別子“04”が格納された後には、定常的に送受信されるフレームとして、再びノード1、ノード3を経由してノード2に戻り、ノード2はこのフレームの内容によってリングトポロジテーブルの設定内容を認識する。

【0025】ノード4では、ノード2側から第1バイト目に“02”が格納されたフレームと、“01”が格納されたフレームとを受け取り、第1バイト目に“02”が格納されたフレームの第2バイト目に自ノード識別子“04”を格納し、また第1バイト目に“01”、第2バイト目に“03”、第3バイト目に“02”が格納されたフレームの第4バイト目に自ノードの識別子“04”を格納して、それぞれノード1に対して送出する。そして、第1バイト目に“01”が格納されたフレームが、再びノード1、3、および2を経由して自ノードに戻ってきた時点で、リングトポロジテーブルの設定内容を認識する。

【0026】ノード1では、ノード4から初めて送出されるフレーム、すなわち第1バイト目に“04”が格納されたフレームと、ノード2から初めて送出され、ノード4を介して送られたフレーム、すなわち第1バイト目に“02”、第2バイト目に“04”が格納されたフレームとをそれぞれ廃棄し、自ノードから初めて送出したフレーム、すなわち第1バイト目に“01”が格納されたフレームがネットワークを一周して自ノードに戻ってきた時点で、リングトポロジテーブルの設定内容を認識する。

【0027】すなわち、図4においては、各ノードが図3に示した処理を実行することによって、第1バイト目に最小のノード識別子、すなわち“01”が格納されたフレームのみが定常的に送受信されるフレームとして残り、リング内をスルーに回り続けることになる。そして、このフレームの内容によってリング状のネットワークのトポロジ、すなわちノードの並びを各ノードにおいて認識することが可能となる。反時計回りに対しても同様の手順によって反時計回りのリングトポロジを認識することが可能となり、各ノードは時計回りと反時計回りのトポロジを比較することによって、リングトポロジ認識結果の正当性をチェックすることが可能となる。

【0028】なお、ここでは最小のノード識別子が第1バイト目に格納されたフレームが残るものとして第1の実施例を説明したが、逆に受信フレームの第1バイト目が自ノードの識別子より小さい時に受信フレームを廃棄することとすれば、結果として第1バイト目に最大のノード識別子を格納したフレームのみが定常的なフレームとしてネットワーク内をスルーに回ることになる。

【0029】図3および図4で説明したように、最小の

ノード識別子が第1バイト目に格納されたフレームのみがネットワーク内をスルーに回るフレームとして残る場合に、仮にその最小の識別子を持つノードが2つネットワーク内に存在した場合には、正しいリングトポロジテーブル構築処理が不可能となる。図5はそのような場合をチェックするための最小ノード識別子重複検出処理のフローチャートである。

【0030】同図において、処理が開始されると、まずステップS10で各ノードがスルーしていく受信フレームを常にモニタし、ステップS11で常に同じフレームを受信しているか否かを判定し、常に同じフレームである場合には何らの処理を行うことなく、また異なるフレームが存在する時にはステップS12でノード識別子重複アラームを発生して処理を終了する。

【0031】図6はこのように最小のノード識別子が2つ存在する場合の通信ネットワークの例である。同図において、最小のノード識別子として①を持つノード(AとD)が2つ存在する。この場合には、ネットワーク内に同一のノード識別子があると処理を抜けてフレームをスルーさせるため、図3の処理によってネットワーク内でスルーに回るフレームとしてノード識別子の並びが①、②、③となるフレーム(ノードDで受信したフレームをスルーさせる)と、①、④、⑤となるフレーム(ノードAで受信したフレームをスルーさせる)との2種類が存在することになる。このようにスルーで回る受信フレームを各ノードがモニタすることにより図5のステップS12で重複アラームが発生することにより、最小ノード識別子の重複を検出することができる。

【0032】図7は図5と異なる最小ノード重複検出処理のフローチャートである。同図は図3の処理を時計回りと反時計回りの両方の方向に対して実行し、時計回りと反時計回りのフレームがネットワーク内を送受信される場合の最小識別子ノード重複検出処理のフローチャートである。

【0033】図7において、ステップS15でそれぞれのノードは自ノードの両側のノードからフレームを受信し、ステップS16で反時計回り方向にネットワークを回る受信フレーム内部のノード識別子格納領域内のノード識別子を自ノードの識別子を先頭にして並び替え、ステップS17で時計回りと反時計回りの両方から受信したフレーム内の識別子の並びが同じか否かを判定し、両者が異なる場合にはステップS18でノード識別子重複を示すアラームを発生し、同じ場合には何らの処理を行うことなく処理を終了する。

【0034】次に、本発明の第2の実施例について説明する。第2の実施例では通信ネットワーク内で張られている通信回線、すなわちチャネルのそれぞれに対して、そのチャネルの送信元ノードと受信先ノードとを示す情報が設定される回線情報テーブルが各ノードにおいて作成される。図8は第2の実施例における回線情報テーブ

ル作成の概念の説明図である。

【0035】図8において、例えばノード3において東側、すなわちノード2からのE-W回線（時計回り）がドロップ設定されている、すなわちノード3がその回線のデータの受信先となっていることと、またノード3がノード4へのE-W回線に対してアッド設定されている、すなわちノード3がその回線の送信元ノードとなっていることを仮定して、ノード3でのE-W回線に対する回線情報テーブルの設定について説明する。

【0036】まず、ノード4に向かう方向のE-W回線に対するW側回線情報テーブルの内容設定について説明する。前述のように、このチャンネルはノード3でアッド設定されているために、ノード3は自ノードの識別子（またはアドレス）をE-W方向のソース識別子（ソースアドレス）としてW側回線情報テーブルに登録すると共に、後述するE-Wソースフレームにソース識別子として挿入し、そのフレームをノード4側に送り出す。また、ノード4から、例えばW-E回線（反時計回り）を経由して受け取ったE-Wディスティネーションフレームからディスティネーション識別子（ディスティネーションアドレス）を取り出し、その識別子をE-Wディスティネーション識別子としてW側回線情報テーブルに登録する。

【0037】次に、E側回線情報テーブルの作成について説明する。ノード2からのE-W回線はノード3においてドロップ設定されているために、ノード3は自ノードの識別子をE-W方向のディスティネーション識別子としてE側回線情報テーブルに登録すると共に、例えばW-E回線を介してノード2に送るディスティネーションフレームにそのディスティネーション識別子を挿入する。また、ノード2から送られるE-Wソースフレームからソース識別子を取り出し、その識別子をE-Wソース識別子として自ノードのE側回線情報テーブルに登録する。

【0038】以上の動作によって、E-W方向の信号についてソース局とディスティネーション局の両方において、回線情報テーブルを構築することができる。また、W-E方向の信号についても同様の動作を行うことによって、ノード3における全回線情報を示す回線情報テーブルが作成される。

【0039】図9はソースフレームとディスティネーションフレームにおけるソースアドレス（ソース識別子）とディスティネーションアドレス（ディスティネーション識別子）の格納方法の説明図である。同図において、1つのフレームの中に存在するソースアドレスフィールドとディスティネーションアドレスフィールドだけが示されており、これらのフレームではそれぞれのアドレスに8ビットずつが割り当てられ、チャンネル1からチャンネルnの各チャンネルに対するフレームにソースアドレスやディスティネーションアドレスが格納されることが示さ

れている。なお、図9においてTX側は送信側、PX側は受信側を示し、またソースフレームとディスティネーションフレームとの間にはフレーム自体の区別は存在しない。

【0040】すでに図8に関して一部説明したが、これらのソースフレームおよびディスティネーションフレームとしては①E-Wソースフレーム、②E-Wディスティネーションフレーム、③W-Eソースフレーム、および④W-Eディスティネーションフレームの4種類があることになり、時計回りの回線、すなわちE-Wの回線には①および④のフレームが、また反時計回り、すなわちW-Eの回線には②および③のフレームが乗せられて、それぞれの方向に送られる。送られたフレームに対しては、各ノードにおいて各チャンネルの各フレーム毎に単に中継するだけのスルー、または送信元ノードであることを示すアッド設定、受信先ノードであることを示すドロップ設定を、通信の状態に応じて自由に設定することができる。そして、各ノードはこれらのフレームに識別子を格納したり、あるいは取り出したりすることによって、全ての回線の全てのチャンネルに対してソース識別子とディスティネーション識別子とを認識し、回線情報テーブルを構築することが可能となる。

【0041】図10は自ノードがアッド設定、またはドロップ設定となっている回線に対する回線情報テーブル作成処理のフローチャートである。このフローチャートは回線の1つの方向、例えば右回り方向のみに対する処理を示すが、反対方向に対する作成処理も同様である。またこの処理は各チャンネル毎に行われる。

【0042】図10において、ステップS20で電源がオンとされるか、または他の要因で処理が開始されると、自ノードの回線設定の変更処理などが行われる。例えば、自ノードにおいて自明であるような回線情報の回線情報テーブルへの格納もこのステップで行われる。その後、ステップS21でネットワークからのフレームが受信され、ステップS22で受信したフレームのうちTX側のフレームのソースアドレスと、RX側のフレームのディスティネーションアドレスに自ノードの識別子を入れた送信フレームが作成される。

【0043】続いてステップS23で、作成された送信フレームと受信したフレームとの間で該当チャンネルのデータが比較され、不一致の時にはS24で送信フレームが送信された後に、S21以降の処理が繰り返される。ステップS23でデータが一致した時にはステップS25で該当チャンネルの回線情報テーブルの作成終了と判定され、処理を終了する。なお、ステップS22、およびS23における処理の詳細については後述する。

【0044】図11は自ノードが通信ネットワークにおけるある回線の中継ノードとなっている場合、すなわちスルー設定されている場合の回線情報テーブル作成処理フローチャートである。同図においてステップS27で

電源オンなどで処理が開始され、ステップS28で自ノードがスルー設定となっていることが確認され、S29で回線上のフレームがモニタされ、それぞれの回線のソースアドレスとディスティネーションアドレスとが抽出されて、自ノードの回線情報テーブルに格納される。

【0045】図12は回線情報テーブルの作成の具体例の説明図である。同図においてはノード2のW側と、ノード3のE側の回線情報テーブルの作成法が説明されている。

【0046】まず、図12(a)においてノード2はE-W方向の回線に対してアド設定されているために、E-W方向のW側テーブルのソース識別子として自ノードの識別子を格納し、その識別子をE-Wソースフレームに挿入してノード3に送り出す。ノード3ではこのソースフレームを受けてソース識別子を取り出し、これをE側テーブルのE-W方向のソース識別子として格納する。一方、ノード3はE-W方向の回線が自ノードでドロップされているために、自ノードの識別子をE側テーブルのE-W方向のディスティネーション識別子として格納し、その識別子をE-W方向のディスティネーションフレームに挿入し、ノード2に送り出す。ノード2ではこのディスティネーションフレームの内容からディスティネーション識別子を取り出し、W側テーブルのE-W方向に対するディスティネーション識別子として格納する。

【0047】次に、W-E方向について説明する。ノード3はこの方向の回線が自ノードに対してアド設定されているために、E側テーブルのW-E方向に対するソース識別子として自ノードの識別子を格納し、その識別子をW-Eソースフレームに挿入して、ノード2に送出する。ノード2ではこのフレームからソース識別子を取り出し、W側テーブルのW-E方向に対するソース識別子として格納する。ノード2はこの回線が自ノードにおいてドロップ設定されているために、自ノードの識別子をW側テーブルのW-E方向のディスティネーション識別子として格納すると共に、その識別子をW-Eディスティネーションフレームに挿入し、ノード3に送り出す。ノード3ではこのフレームからディスティネーション識別子を取り出し、E側テーブルのW-E方向のディスティネーション識別子として格納する。

【0048】次に図12(a)の場合を例として、図10におけるステップS22、S23の処理の具体例を説明する。図12(b)は、図12(a)においてノード2からノード3に送られるE-WソースフレームとW-Eディスティネーションフレームを示す。ノード2においては、E-W方向では自ノードがアドに設定されているために、E-W方向のソースアドレスとして自ノードの識別子、またW-E方向では自ノードがドロップに設定されているためにW-E方向のディスティネーションアドレスとして自ノードの識別子を格納し、ノード3側に

送信する。

【0049】図12(c)はノード3側からノード2側に送り返されるE-Wディスティネーションフレームと、W-Eソースフレームを示す。ノード3では自ノードがE-W方向の受信先ノードであり、W-E方向の送信元ノードであることから、E-Wディスティネーションフレームに自ノードの識別子を、またW-Eソースフレームに自ノードの識別子を格納して、ノード2に対してこれらのフレームを送信する。

【0050】図12(c)が図10のステップS21で受信したフレームに相当し、ノード2ではステップS22でこの受信フレームに自ノードの識別子を格納し、ノード3に送信しようとするが、この時点でステップS23で自ノードから送信しようとするフレームと受信したフレームとの該当データが一致しているために、このチャネルに対する回線情報テーブルの構築が終了したことを認識することになる。

【0051】図12では、ノード2とノード3の間で双方向の信号伝送区間が一致している場合について回線情報テーブルの作成処理を説明したが、一般に通信ネットワークにおいて双方向の信号伝送区間が一致しているとは限らず、両者が異なってくることも多い。また、信号伝送区間中のあるノードが、その回線に対する送信元にも受信先にもならず、単に信号を中継するだけのことも多い。図13はそのような場合に対応する回線情報テーブル作成処理の説明図である。

【0052】図13（ノードの内容については図14参照）においてW-E方向の回線はノード1が送信元ノードとなっており、ノード4とノード3は単に回線の中継するのみに留まり、受信先ノードはノード2である。それに対して、E-W方向の回線はノード2からノード3の間で1つ張られ、またノード3からノード1までノード4を中継する形でもう1つ張られた形式となっている。

【0053】E-W方向の回線を対象として、各ノードにおけるテーブル作成処理について個別に説明する。ノード1では、W側でデータの送受信が行われていないために、E側の回線情報テーブルの作成処理のみが行われる。まず、ノード4側から送られたE-Wソースフレームからソース識別子、ここでは“3”が抽出され、E側テーブルのE-W方向のソース識別子として格納される。次に、E-W回線は自ノードがドロップとなっているために、自ノードの識別子をE側テーブルのE-W方向のディスティネーション識別子として格納すると共に、E-Wディスティネーションフレームにその識別子を挿入してノード4側に送り出す。

【0054】ノード2ではE側ではデータの送受信が行われていないので、W側のテーブルのみが作成される。E-W方向の回線に対しては自ノードがアドに設定されているため、自ノードの識別子をW側テーブルのE-

W方向のソース識別子として格納すると共に、E-Wソースフレームに自ノードの識別子を挿入してノード3に送り出す。次に、ノード3側から送られたE-Wディスティネーションフレームからディスティネーション識別子、ここでは“3”を取り出し、W側テーブルのE-W方向のディスティネーション識別子として格納する。

【0055】ノード3はW-E方向の回線が中継のみ、すなわちスルーに設定されており、ノード2からのE-W回線はドロップに設定され、ノード4へのE-W回線はアッドに設定されている。このためE側、およびW側の両方向のテーブルが作成される。

【0056】まず、E側のテーブル作成処理について説明する。E-W回線を経由してノード2から送られたE-Wソースフレームからソース識別子として、ここでは“2”が取り出され、E側テーブルのE-W方向のソース識別子として格納される。また、ノード3は自ノードがE-W方向に対しドロップ設定されていることから、E側テーブルのE-W方向のディスティネーション識別子として自ノードの識別子を格納し、その識別子をE-Wディスティネーションフレームに挿入してノード2に送り出す。

【0057】次に、W側テーブルについては、自ノードがW方向にアッド設定されていることから、W側テーブルのE-W方向のソース識別子として自ノードの識別子を格納し、その識別子をE-Wソースフレームに挿入してノード4側に送り出す。また、ノード4から送られるE-Wディスティネーションフレームからディスティネーション識別子、ここでは“1”を取り出し、W側テーブルのE-W方向のディスティネーション識別子として格納する。

【0058】最後にノード4におけるテーブル作成処理を説明する。ノード4はE-W方向、W-E方向共にスルーに設定されているために、スルーに通過するソースフレームとディスティネーションフレームからノード識別子を取り出して、それぞれのテーブルに設定する処理のみが行われる。例えば、E側ではノード3から送られるE-Wソースフレームからソース識別子、ここでは“3”が取り出され、E-W方向のソース識別子として格納される。また、ノード1から送られ、自ノードを中継してノード3に送られるE-Wディスティネーションフレームからノード識別子、ここでは“1”が取り出され、E-W方向のディスティネーション識別子として格納される。W側テーブルの作成処理もほぼ同様であるので、説明を省略する。

【0059】図15は図13に対してW-E方向についても回線情報テーブルの作成した最終結果を示している。W-E方向のテーブル作成処理は図13におけるE-W方向のテーブル作成処理とほぼ同様であるので、詳細な説明は省略する。また、以上の説明においては1つのチャンネルのみを対象として回線情報テーブルの作成処

理を説明したが、実際に通信ネットワークにおいて多数のチャンネルが用いられている場合にも各チャンネルに対して同様の処理が個別に行われる。チャンネル相互間に相関々係は存在しない。

【0060】以上において第1の実施例としてネットワークのトポロジ、ここではリングトポロジテーブルの作成処理について、また第2の実施例では通信ネットワーク内の各回線、すなわちそれぞれのチャンネルの送信元ノードと受信先ノードとの情報を示す回線情報テーブルの作成について詳細に説明した。本発明においては、これらのテーブルを各ノードにおいて自動的に作成することが可能になる。そして、これらのテーブルの内容は通信ネットワーク内の情報として各種の場合に有効に利用されるが、その1つが障害発生時の利用である。図16、および図17はこの障害発生時における利用法の説明図である。

【0061】図16は回線切断時におけるリングトポロジテーブルの利用例の説明図である。同図は(a)のようにネットワーク内でノード3からノード4に対する1つの回線、すなわち1つのチャンネルのみが張られている場合に、(b)のようにノード3からノード4への回線が切断した場合の回線救済方法の説明図である。

【0062】この回線切断の障害はノード4の受信側において受信信号の断として検出され、その切断情報はノード3に送られ、これによってノード3側では回線切断を検出する。そして、この回線切断に応じて、ノード3からノード2、ノード1を経由してノード4に至る経路を用いて回線復旧が行われる。回線切断によってどのノードとどのノードの間の回線が切断したのか、どのノードを経由して受信先ノード、ここではノード4への通信を行えばよいかなどがリングトポロジテーブルの内容を基にして判定されることになる。

【0063】図17は回線情報テーブルの設定内容の利用法の説明図である。回線情報テーブルは、図16のように障害の救済自体には用いられず、複数障害が発生した場合のミスコネクトの検出に用いられる。

【0064】図17(a)において、同一のチャンネルがノード3からノード4、ノード4からノード1の間でそれぞれ独立に通信に使用されているものとする。図17(b)に示すように、これらの回線、すなわちノード3からノード4の間と、ノード4からノード1の間との両方に回線切断が発生したものとする。図16で説明した方法では、ノード4からノード1に張られていた回線はノード3からノード4に張られていた回線と同じであるので、ノード3からノード2を経由してノード1に至る救済回線が設定されてしまう。この救済回線は、ノード1にとってはノード4からの通信を回復するものではなく、ミスコネクトを起こしてしまったことになる。

【0065】しかしながら、回線情報テーブルの設定内容を利用することによって、ノード1ではこの回線を介

して自ノードが受信する信号はノード4からのものであることを知ることができ、リングが2個所で切断されてしまったことを認識することができる。そして、このミスコネクトに対応してアラーム信号を発生することができ、ミスコネクト状態を解消することができる。このような処理に回線情報テーブルの設定内容が利用される。

【0066】

【発明の効果】以上詳細に説明したように、本発明によればネットワーク構成、すなわちノードの並びを各ノードで自動的に認識することができ、またネットワーク内での各回線、すなわち各チャネルがどのノードからどのノード宛のものであるかという回線情報を各ノードで認識することが可能となる。そして各ノードでは、例えば障害発生時にこれらの内容を利用して回線復旧を行ったり、ミスコネクトを検出することができ、通信ネットワークにおける実際の通信状態にあった回線情報の利用が可能となる。

【図面の簡単な説明】

【図1】本発明の第1の実施例に対する機能ブロック図である。

【図2】本発明の第2の実施例に対する機能ブロック図である。

【図3】第1の実施例におけるリングトポロジテーブル作成処理のフローチャートである。

【図4】リングトポロジテーブル作成処理の具体例の説明図である。

【図5】最小ノード識別子重複検出処理のフローチャートである。

【図6】ネットワークにおけるノードの並びの具体例の説明図である。

【図7】図5と異なる最小ノード識別子重複検出処理のフローチャートである。

【図8】回線情報テーブル作成処理の概念の説明図である。

【図9】フレーム内のソースアドレスとディスティネーションアドレスの格納方法の説明図である。

【図10】回線情報テーブル作成処理のフローチャート（その1）である。

【図11】回線情報テーブル作成処理のフローチャート（その2）である。

【図12】双方向の通信区間が同一の場合の回線情報テーブル作成処理の具体例の説明図である。

【図13】双方向の通信区間が同一でない場合の回線情報作成処理の具体例の説明図（その1）である。

【図14】図13におけるノード（NODE）1～4の内容を示す図である。

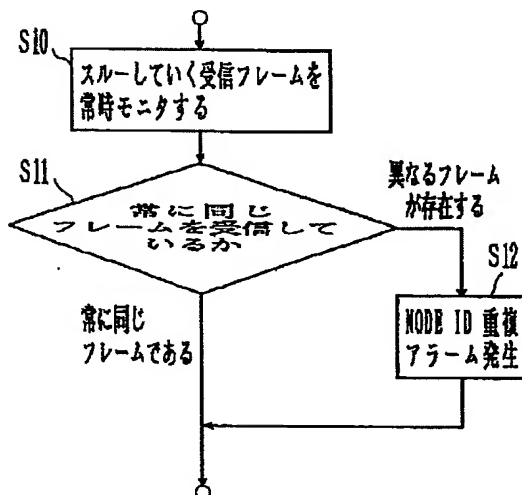
【図15】双方向の通信区間が同一でない場合の回線情報作成処理の具体例の説明図（その2）である。

【図16】回線切断時におけるリングトポロジテーブルの利用方法の説明図である。

【図17】複数障害が発生した場合の回線情報テーブルの利用方法の説明図である。

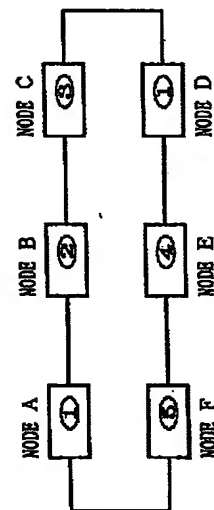
【図5】

最小ノード識別子重複検出処理のフローチャート



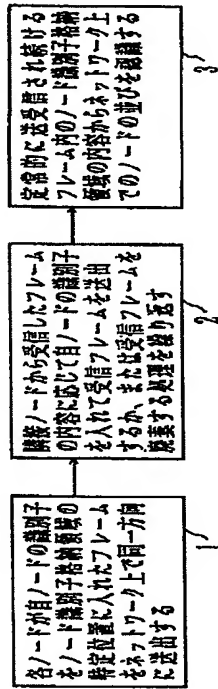
【図6】

ネットワークにおけるノードの並びの
具体例の説明図



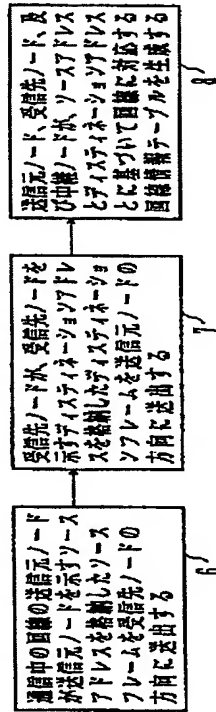
【図 1】

本発明の第1の実施例に対する
機能ブロック図



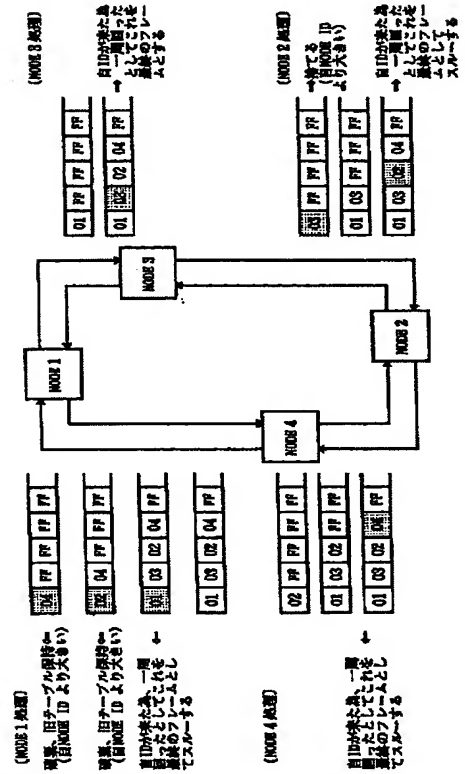
【図 2】

本発明の第2の実施例に対する
機能ブロック図



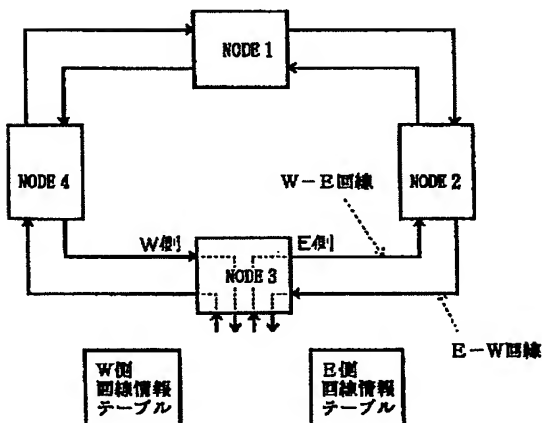
【図 4】

リングトポロジカル作成処理の
具体例の説明図



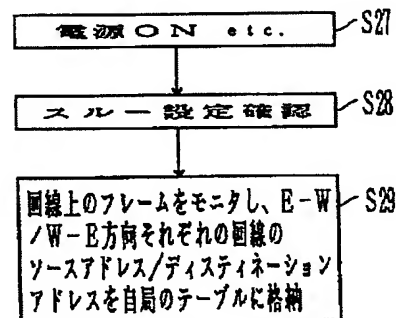
【図 8】

回線情報テーブル作成処理の概念の説明図



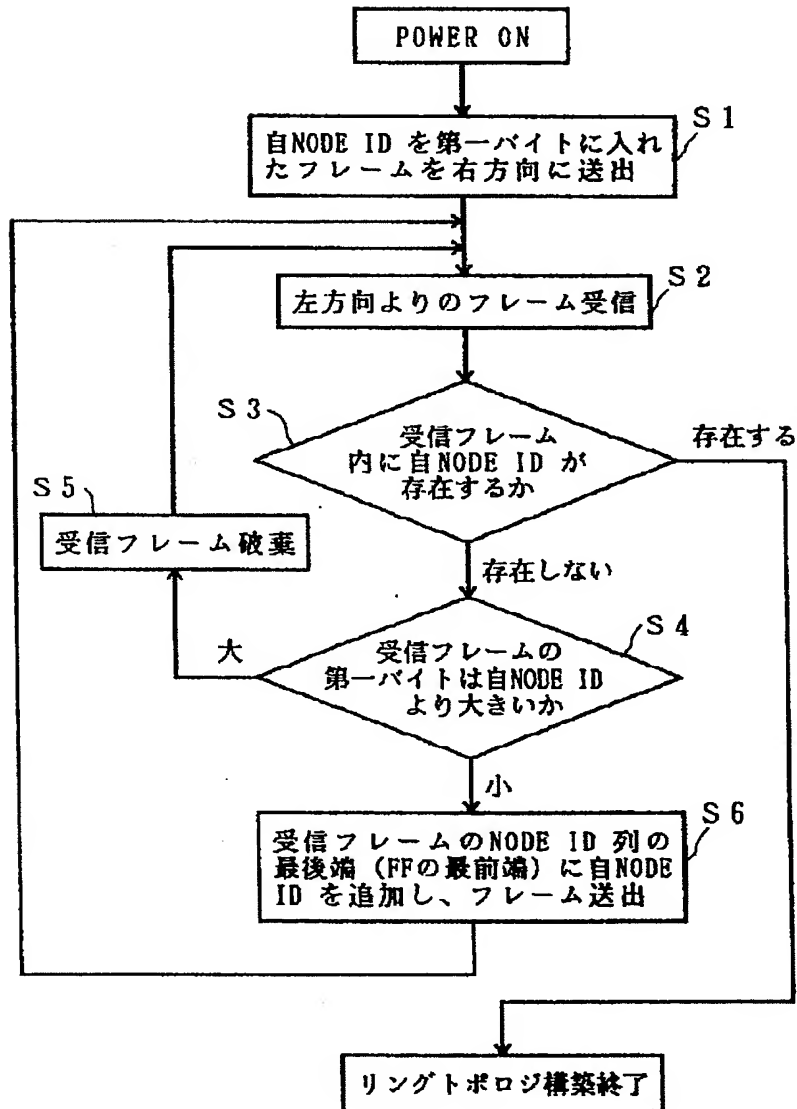
【図 11】

回線情報テーブル作成処理のフローチャート
(その2)



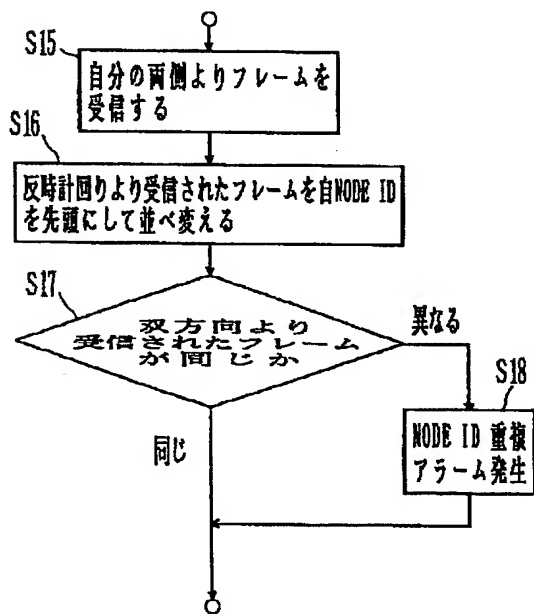
【図3】

第1の実施例におけるリングトポロジテーブル
作成処理のフローチャート



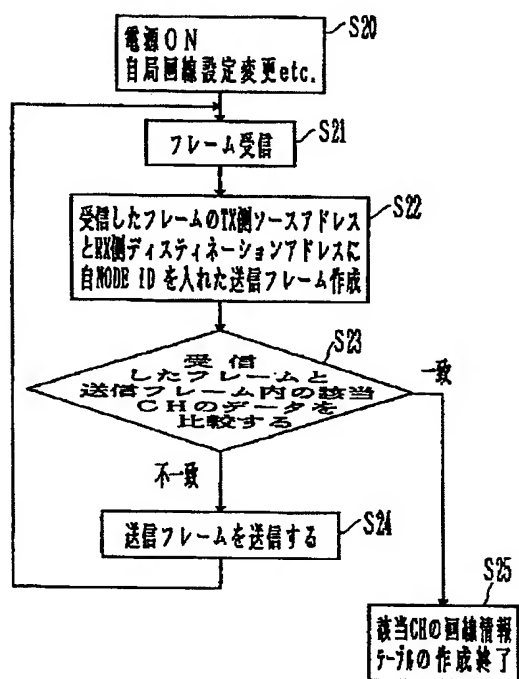
【圖 9】

フレーム内のソースアドレスとディスティネーションアドレスの格納方法の説明図



【図 10】

回線情報テーブル作成処理のフローチャート
(その1)

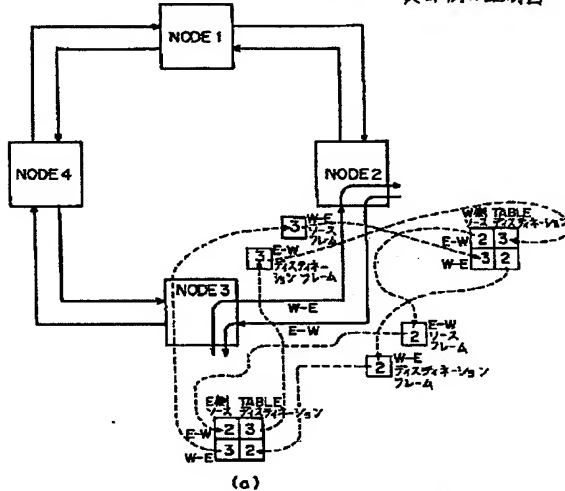


CH1	TX側	ソースアドレス	ディクタイネーシム	ソードレス
	RX側	ソースアドレス	ディクタイネーシム	ソードレス
CH1	TX側	ソースアドレス	ディクタイネーシム	ソードレス
	RX側	ソースアドレス	ディクタイネーシム	ソードレス

CHn	TX側	ソースアドレス	ディクタイネーシム	ソードレス
	RX側	ソースアドレス	ディクタイネーシム	ソードレス

【図 12】

双方向の通信区画が同一の場合の回線情報テーブル作成処理の
具体例の説明図



(a)

	ソースアドレス	デスティネーションアドレス
TX(E-W)	2	未定
RX(W-E)	未定	2

(b)

	ソースアドレス	デスティネーションアドレス
TX(E-W)	2	3
RX(W-E)	3	2

(c)

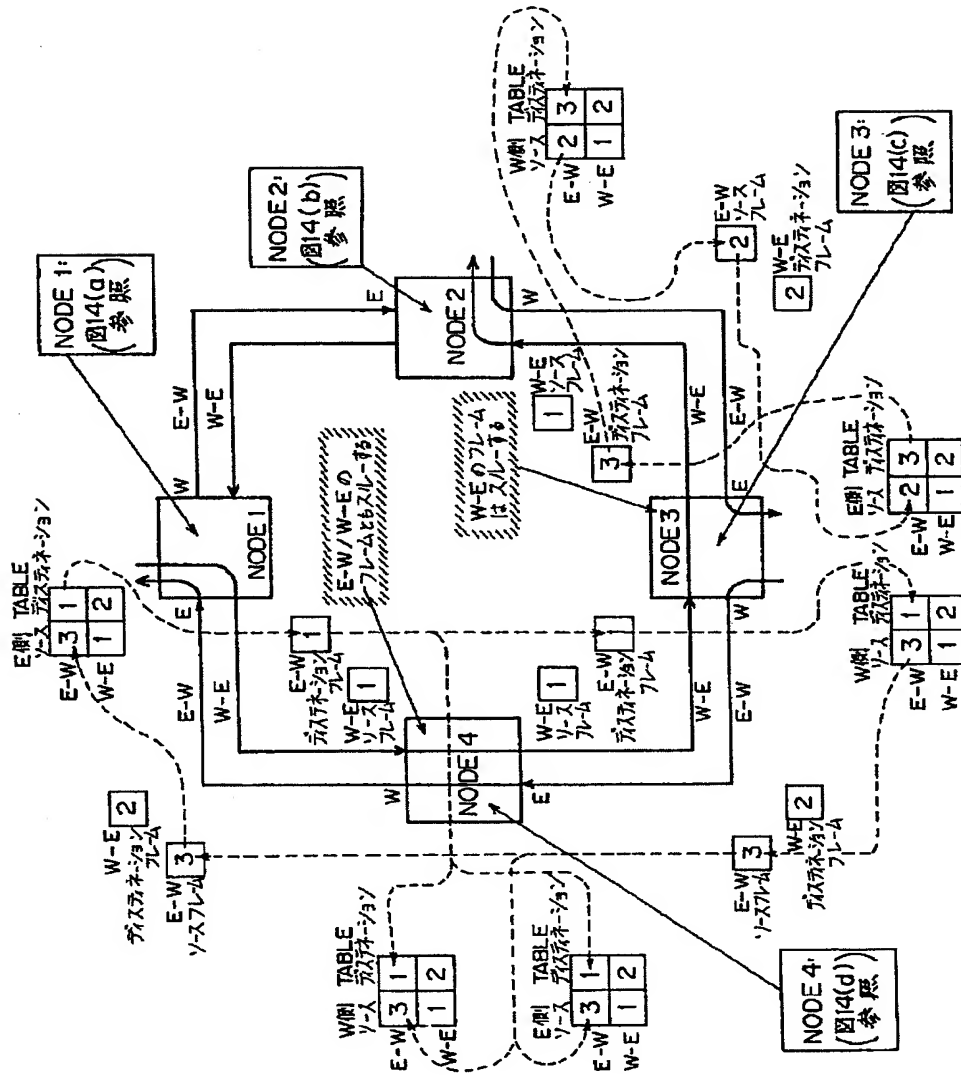
【図 14】

図 13 におけるノード (NODE) 1~4 の内容を示す図

- (a) NODE 1:
- E 例
 - ① E-WソースフレームよりE-W回線のソースIDを抽出し、テーブルに登録する。
 - ② E例がDROPにクロスコネクト設定されているので、E-W回線は自NODEがデスティネーションであると認識し、自テーブルに登録するとともに、E-Wデスティネーションフレームに自NODE IDを入れる。
 - W 例
 - ① クロスコネクトされていないので処理無し
- (b) NODE 2:
- E 例
 - ① クロスコネクトされていないので処理無し
 - W 例
 - ① クロスコネクト情報より自ノードでADDされているので、自ノードIDをE-WのソースIDとして登録し、また、E-Wソースフレームの該当CH群に入れる。
 - ② 対局からのデスティネーションフレーム内のIDを読み取り、デスティネーションIDとして登録する。
- (c) NODE 3:
- E 例
 - ① E-WソースフレームよりE-W回線のソースIDを抽出し、テーブルに登録する。
 - ② E例がDROPにクロスコネクト設定されているのでE-W回線は自NODEがデスティネーションであると認識し、自テーブルに登録するとともに、E-Wデスティネーションフレームに自NODE IDを入れる。
 - W 例
 - ① W例がADDにクロスコネクト設定されているのでE-W回線は自NODEがソースであると認識し、自テーブルに登録するとともに、E-Wソースフレームに自NODE IDを入れる。
 - ② E-WデスティネーションフレームよりIDを抽出し、デスティネーションIDとして登録する。
- (d) NODE 4:
- E 例
 - ① E-WソースフレームよりE-W回線のソースIDを抽出し、テーブルに登録する。
 - ② E-W回線はスルーにクロスコネクト設定されているのでE-Wソースフレームをスルーさせる。
 - ③ E-W回線はスルーにクロスコネクト設定されているのでスルーしているE-WデスティネーションフレームよりE-WデスティネーションIDとして登録する。
 - W 例
 - ① E-W回線はスルーにクロスコネクト設定されているのでE-WソースフレームよりE-W回線のソースIDを抽出し、テーブルに登録する。
 - ② E-WデスティネーションフレームよりIDを抽出し、デスティネーションIDとして登録する。
 - ③ E-W回線はスルーにクロスコネクト設定されているのでE-Wデスティネーションフレームをスルーさせる。

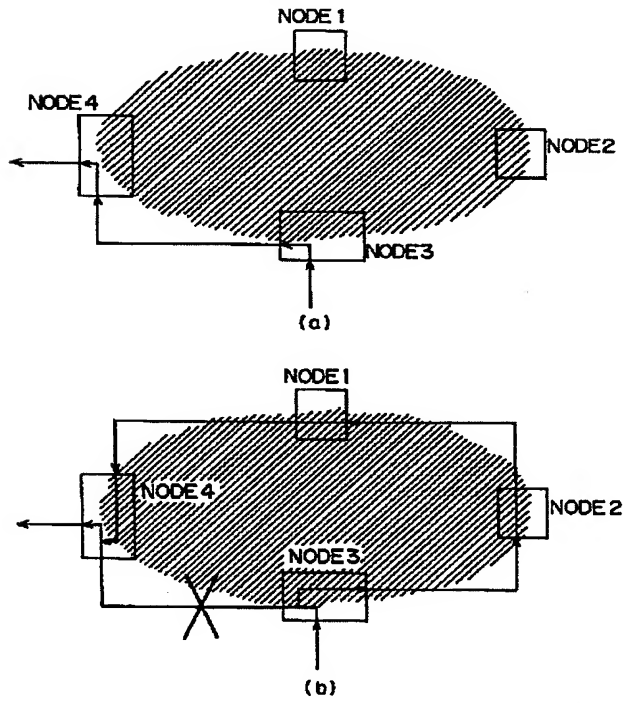
【図13】

双方向の通信区間が同一でない場合の回線情報作成処理の
具体例の説明図(その1)



【図 16】

回線切断時におけるリングトポロジテーブルの
利用方法の説明図



【図 17】

複数障害が発生した場合の回線情報テーブルの
利用方法の説明図

